**Distribution Centre – Database Report**

## **Database Description**

When it comes to storing goods, we might think it is as simple as renting a building and turning it into a warehouse. But, the type of storage space we select can have an impact on our ability to satisfy the customers by fulfilling the orders on time. Therefore, it is very important to spend some time thinking which type of warehouse we need to support the business’s objectives.

There are 6 different types of warehouse, distribution centre, climate-controlled warehouse, private warehouse, public warehouse, automated warehouse and fulfilment centre. In this project I have chosen to implement a database of a distribution centre.

Distribution warehouses are used to store and sell large quantities of goods. It typically houses goods from multiple manufactures/suppliers and sell them to the retailers. Goods are usually housed for short period of the time, sometimes just a day at a time. To run the supply chain smoothly they need to be able to serve a high volume of customers and house large amount of goods specially around the holiday seasons.

The database mainly describes the employees, equipment, suppliers, customers, delivery, storage and the products. I used staff, equipment, courier, customer, storage/shelf, product and supplier as a table to describe the warehouse. All the tables have an identifier called primary key. Some of the information from the tables references to another by a foreign key.

The staff table contains the staff\_id, staff\_name, date\_of\_birth, staff\_address and job\_id. Staff\_id is the primary key and the foreign key is job\_id which is referenced to storage\_id in storage/shelf table.

The equipment table contains the equipment\_id, equipment\_name, equipment\_type and user\_id. Equipment\_id is the primary key and the foreign key is user\_id which is references to staff\_id in staff table.

The supplier table contains supplier\_id, supplier\_name, delivery\_item, phone\_num and customer\_id. Supplier\_id is the primary key and the foreign key is customer\_id which is references to customer\_id in customer table.

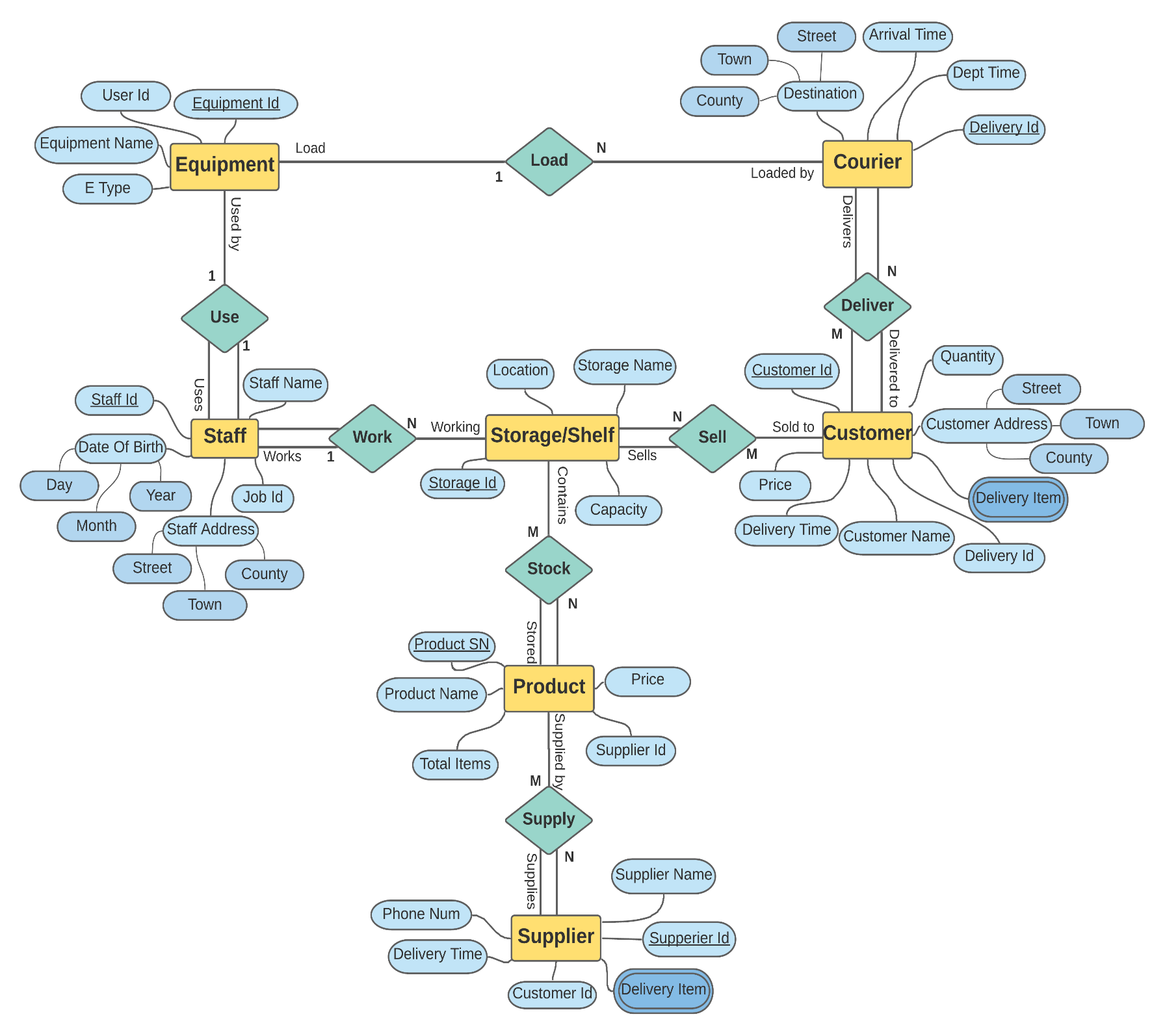
The courier table contains the delivery\_id, destination, dept\_time and arrivial\_time. Delivery\_time is the primary key of the table.

The customer table contains the customer\_id, customer\_name, customer\_address, delivery\_item, delivery\_id, delivery\_time, price and quantity. Customer\_id is the primary key of the table.

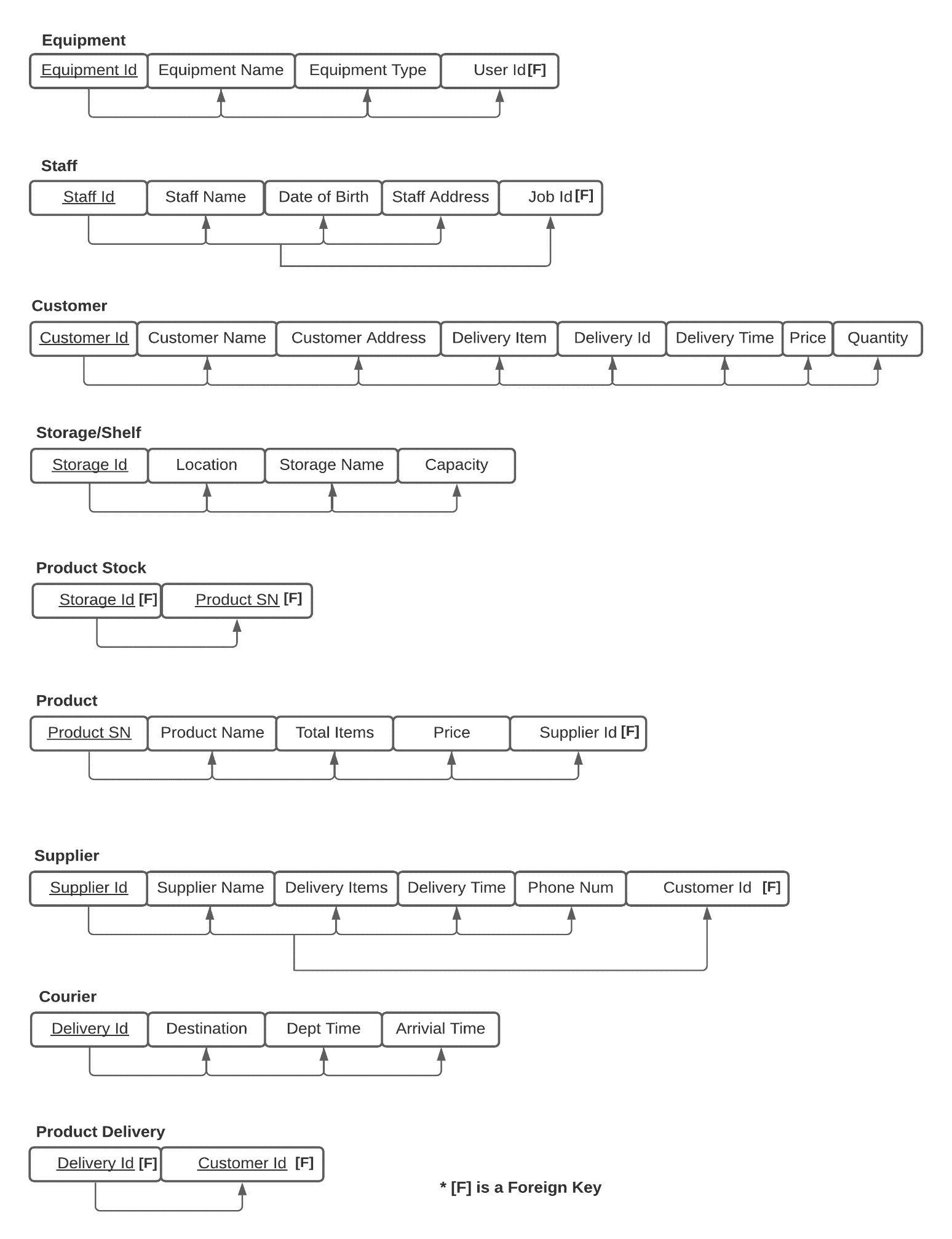
The storage/shelf table contains the storage\_id, location, storage\_name and the capacity. Storage\_id is the primary key of the table.

The product table contains the product\_sn, product\_name, total\_items, price and supplier. Product\_sn is the primary key and the foreign key is supplier\_id which references to supplier\_id in supplier table.

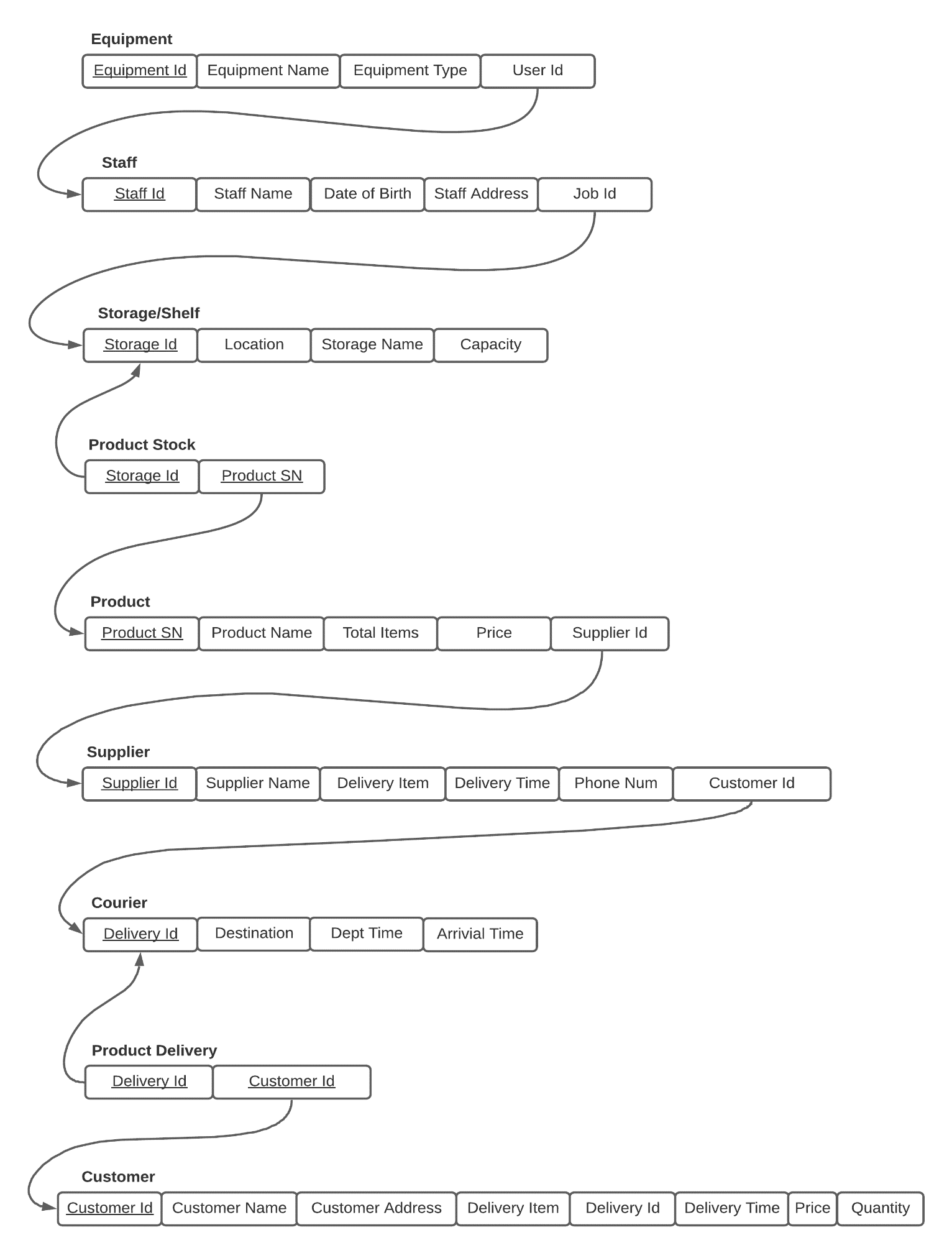
## **Entity Relationship Diagram**

The entity relationship diagram for the database is shown below:

## **Functional Dependency Diagram**

The functional dependency diagram of the database is shown below:

## **Mapping to Relational Schema**

The mapping from the entity relationship diagram to an outline relational schema is shown below:

## **Implicit Constraints**

**Equipment**

Primary Key:

Equipment\_Id VARCHAR (10) not null

Foreign Key(s):

Foreign Key (User\_Id) References Staff (Staff\_Id)

**Staff**

Primary Key:

Staff\_Id Integer not null

Foreign Key(s):

Foreign Key (Job\_Id) References Shelf (Storage\_Id)

**Storage/Shelf**

Primary Key:

Storage\_Id Varchar (20) not null

Foreign Key(s):

Null

**Product**

Primary Key:

Product\_SN INTEGER not null

Foreign Key(s):

Foreign Key (Supplier\_Id) References Supplier (Supplier\_Id),

**Supplier**

Primary Key:

Supplier\_Id INTEGER not null

Foreign Key(s):

Foreign Key (Customer\_Id) References Courier (Delivery\_Id),

**Courier**

Primary Key:

Delivery\_Id Integer not null

Foreign Key(s):

Null

**Customer**

Primary Key:

Customer\_Id Integer not null

Foreign Key(s):

Null

## **Semantic Constraints**

There are many semantic constraints in this database. Very first constraint is in the customer identification number, it can only be 8 digit long.

CONSTRAINT Check\_Customer\_Id CHECK(LENGTH(Customer\_Id) = 8)

There is also a constraint on staff’s job\_id. It checks if the job\_id given to the staff exists or not.

CONSTRAINT Check\_Job\_Id CHECK (Job\_Id IN ('Elec0001', 'OutAcc0501', 'MuscDJ0007', 'HFHA0001', 'FaClAc01', 'HeBe001', 'GaToy01', 'HoFur123')

## **Triggers**

One of the triggers occurs when a employee leaves the company and new person is hired. New employee will continue in the same position as the last employee.

CREATE OR REPLACE TRIGGER remove\_staff\_update\_equipment

AFTER DELETE ON Staff

FOR EACH ROW

WHEN (OLD.Staff\_Id IS NOT NULL)

DECLARE

Del\_Staff\_Id NUMBER;

BEGIN

Del\_Staff\_Id := : OLD.Staff\_Id;

UPDATE Equipment

SET User\_Id = NULL

WHERE User\_Id = Del\_Staff\_Id;

END remove\_staff\_update\_equipment;

.

RUN;

The trigger below occurs when a new employee is hired in the warehouse. His/her id, name, date of birth, address and job id will be inserted to the database.

CREATE OR REPLACE TRIGGER Add\_Staff

AFTER INSERT ON Staff

FOR EACH ROW

DECLARE

Add\_Staff\_Id NUMBER;

BEGIN

Add\_Staff\_Id := :NEW.Staff\_Id;

UPDATE Equipment

SET User\_Id = User\_Id + 1

WHERE User\_Id = Add\_Staff\_Id;

END Add\_Staff;

.

RUN;

## **Update Operations**

There are many update operations which can be performed on this database. Below are some of the commands we can use to update the information on the database.

1. If the employee 0123456 wants to move from Electronics & Office to Musical Instruments & DJ, the information can be updated as following:

UPDATE Staff SET Job\_Id = 'MuscDJ0007' WHERE Staff\_Id = 0123456;

1. If the capacity of the Storage HoFur123(Home and Furniture) is increased from 700 to 1000, the information can be updated as following:

UPDATE Shelf SET Capacity = 1000 WHERE Storage\_Id = 'HoFur123';

## **Relational Select Operations**

Select operation on a database can be performed as shown below:

1. Select all the Staffs in the warehouse:

SELECT \* FROM Staff;

+-------------+------------------------+--------------------+-----------------------------------------------------+------------------+

| Staff\_Id | Staff\_Name | Date\_of\_Birth| Staff\_Address | Job\_Id |

+-------------+------------------------+-------------------+------------------------------------------------------+-------------------+

| 123456 | Roxy Tollemache | 10/02/1990 | 92 Water Rd, Saint Johns, South Dublin | MuscDJ0007 |

| 123457 | Terry Brine | 16/10/1989 | 66 High St, Saint Marks, Kilkenny | OutAcc0501 |

| 123458 | Michael Kennedy | 21/04/1995 | 12 Kill Lane, View park, Waterford | MuscDJ0007 |

| 123459 | Peter Doyle | 27/02/1986 | 47 Kenyon St, Nenagh, Dublin | HFHA0001 |

| 123460 | Alan McCarthy | 02/02/1998 | Duffy Hill, Enniscorthy, South Dublin | FaClAc01 |

| 123461 | Conor Doyle | 07/03/1981 | 104 Creagh Ct., Rathcoole, Dublin | HeBe001 |

| 123462 | Mark Walsh | 16/06/1976 | The Square, Oldcastle, South Dublin | GaToy01 |

| 123463 | Dave Smith | 19/10/1987 | 10 Hanover Qy., Tel, Clondalkin, Dublin | HoFur123 |

+-------------+-------------------------+--------------------+-----------------------------------------------------+-------------------+

1. Select all the products that have over 500 items:

SELECT\*FROM Product WHERE Total\_Items > 500;

+-----------------+----------------------------------+-----------------+------------+------------------+

| Product\_SN | Product\_Name | Total\_Items | Price | Supplier\_Id |

+-----------------+----------------------------------+-----------------+-------------+------------------+

| 1111 | Sony Headphones | 1000 | €219,000 | 123456780 |

| 1112 | Amazon Echo Dot | 1000 | €29,990 | 123456791 |

| 1113 | Tommy Hilfiger Backpack | 1000 | €64,540 | 123456802 |

+------------------+-----------------------------------+----------------+-------------+------------------+

## **Security**

There are many concerns around security of a database. Misuse and unauthorised activities are some of the main issues. To maintain the security of a database there is limited access of the information to the staffs. Only the senior and the manager can remove or update the information. For example, staffs can see who works in which department, but they will not have access to other staff’s home address or date of birth etc. The main objective of this is to hide any sensitive information of the employees.

## **Appendix A – Table Creation**

CREATE DATABASE warehouse;

USE warehouse;

CREATE TABLE Customer (

Customer\_Id Integer not null,

Customer\_Name Varchar (20) not null,

Customer\_Address Varchar (50) not null,

Delivery\_item Varchar (50) not null,

Delivery\_Id Integer not null,

Delivery\_Time Varchar (10) not null,

Quantity Integer not null,

Price Varchar (20) not null,

Primary Key (Customer\_Id),

CONSTRAINT Check\_Customer\_Id CHECK(LENGTH(Customer\_Id) = 8),

CONSTRAINT Check\_Cus\_Delivery\_Id CHECK(LENGTH(Delivery\_Id) = 8)

);

ALTER TABLE Customer

ADD CONSTRAIN Min\_Quantity CHECK (Quantity > 0);

ALTER TABLE Customer add unique (Customer\_Id);

CREATE TABLE Courier (

Delivery\_Id Integer not null,

Destination Varchar (50) not null,

Dept\_Time Varchar (10) not null,

Arrival\_Time Varchar (20) not null,

Primary Key (Delivery\_Id),

CONSTRAINT Check\_Delivery\_Id CHECK(LENGTH(Delivery\_Id) = 8)

);

ALTER TABLE Courier add unique (Delivery\_Id);

CREATE TABLE Shelf (

Storage\_Id Varchar (20) not null,

Location Varchar (20) not null,

Storage\_Name Varchar (50) not null,

Capacity Varchar (20) not null,

Primary Key (Storage\_Id),

CONSTRAINT Quantity\_Check CHECK (LENGTH (Quantity) <= 5000)

);

CREATE TABLE Staff (

Staff\_Id Integer not null,

Staff\_Name Varchar (20) not null,

Date\_of\_Birth Varchar (10) not null,

Staff\_Address Varchar (50) not null,

Job\_Id Varchar (20) not null,

Primary Key (Staff\_Id),

Foreign Key (Job\_Id) References Shelf (Storage\_Id),

CONSTRAINT Check\_Job\_Id CHECK (Job\_Id IN ('Elec0001', 'OutAcc0501', 'MuscDJ0007', 'HFHA0001', 'FaClAc01', 'HeBe001', 'GaToy01', 'HoFur123'))

);

ALTER TABLE Staff add unique (Staff\_Id);

CREATE TABLE Equipment (

Equipment\_Id VARCHAR (10) not null,

Equipment\_Name VARCHAR (50) not null,

Equipment\_Type VARCHAR (50) not null,

User\_Id INTEGER not null,

Primary Key (Equipment\_Id),

Foreign Key (User\_Id) References Staff (Staff\_Id)

);

CREATE TABLE Supplier (

Supplier\_Id INTEGER not null,

Supplier\_Name VARCHAR (50) not null,

Delivery\_item Varchar (50) not null,

Customer\_Id Integer not null,

Delivery\_Time Varchar (10) not null,

Phone\_Num Varchar (20) not null,

Primary Key (Supplier\_Id),

Foreign Key (Customer\_Id) References Courier (Delivery\_Id),

CONSTRAINT Check\_Supplier\_Id CHECK(LENGTH(Supplier\_Id) = 9)

);

CREATE TABLE Product (

Product\_SN INTEGER not null,

Product\_Name VARCHAR (50) not null,

Total\_Items INTEGER not null,

Price VARCHAR (10) not null,

Supplier\_Id INTEGER not null,

Primary Key (Product\_SN),

Foreign Key (Supplier\_Id) References Supplier (Supplier\_Id),

CONSTRAINT Check\_Total\_Items CHECK(LENGTH(Total\_Items) > 0)

);

CREATE TABLE Product\_Stock (

Storage\_Id Varchar (20) not null,

Product\_SN Integer not null,

Primary Key (Storage\_Id, Product\_SN),

Foreign Key (Storage\_Id) References Shelf (Storage\_Id),

Foreign Key (Product\_SN) References Product (Product\_SN)

);

CREATE TABLE Product\_Delivery (

Delivery\_Id INTEGER not null,

Customer\_Id Integer not null,

Primary Key (Delivery\_Id, Customer\_Id),

Foreign Key (Delivery\_Id) References Courier (Delivery\_Id),

Foreign Key (Customer\_Id) References Customer (Customer\_Id)

);

## **Appendix B – Database Population**

INSERT INTO Customer VALUES (12345678, 'PC Universe', '1 Parkview, Greenlanes, Dublin 3', 'Sony Headphones', 87654321, '12:30', 50, '€10,950.00');

INSERT INTO Customer VALUES (12345679, 'Cool Tech', '2 Seaview, Maryland, Dublin 5', 'Amazon Echo Dot', 87654322, '18:00', 30, '€899.70');

INSERT INTO Customer VALUES (12345680, 'Bag Life', '12 Moonwalk, Waterview, Dublin 1', 'Tommy Hilfiger Backpack', 87654323, '15:30', 20, '€1,290.80');

INSERT INTO Customer VALUES (12345681, 'Mulls Music', '53 Sideview, Chelsea Gardens, Dublin 2', 'Wooden Hand Drum', 87654324, '13:30', 15, '€346.35');

INSERT INTO Customer VALUES (12345682, 'Techo', '22 Northsides Park, Swords, Dublin', 'Elgato Stream Deck', 87654325, '11:30', 5, '€1,049.95');

INSERT INTO Courier VALUES (23456789, '1 Parkview, Greenlanes, Dublin 3', '06:00', '12:30');

INSERT INTO Courier VALUES (23456790, '2 Seaview, Maryland, Dublin 5', '06:30', '18:00');

INSERT INTO Courier VALUES (23456791, '12 Moonwalk, Waterview, Dublin 1', '07:30', '15:30');

INSERT INTO Courier VALUES (23456792, '53 Sideview, Chelsea Gardens, Dublin 2', '07:00', '13:30');

INSERT INTO Courier VALUES (23456793, '22 Northsides Park, Swords, Dublin', '08:00', '11:30');

INSERT INTO Shelf VALUES ('Elec0001', '2nd Level', 'Electronics & Office', 5000);

INSERT INTO Shelf VALUES ('Elec0002', '2nd Level', 'Electronics & Office', 1000);

INSERT INTO Shelf VALUES ('OutAcc0501', '1st Level', 'Outdoors & Accessories', 2500);

INSERT INTO Shelf VALUES ('MuscDJ0007', '3rd Level', 'Musical Instruments & DJ', 500);

INSERT INTO Shelf VALUES ('HFHA0001', '4th Level', 'Hi-Fi & Home Audio', 1000);

INSERT INTO Shelf VALUES ('FaClAc01', '3rd Level', 'Fashion, Clothing & Accessories', 5000);

INSERT INTO Shelf VALUES ('HeBe001', '3rd Level', 'Health & Beauty', 5000);

INSERT INTO Shelf VALUES ('GaToy01', '1st Level', 'Gaming & Toys', 4500);

INSERT INTO Shelf VALUES ('HoFur123', '0 Level', 'Home and Furniture', 700);

INSERT INTO Staff VALUES (0123456, 'Roxy Tollemache', '10/02/1990', '92 Water Rd, Saint Johns, South Dublin', 'Elec0001');

INSERT INTO Staff VALUES (0123457, 'Terry Brine', '16/10/1989', '66 High St, Saint Marks, Kilkenny', 'OutAcc0501');

INSERT INTO Staff VALUES (0123458, 'Michael Kennedy', '21/04/1995', '12 Kill Lane, View park, Waterford', 'MuscDJ0007');

INSERT INTO Staff VALUES (0123459, 'Peter Doyle', '27/02/1986', '47 Kenyon St, Nenagh, Dublin', 'HFHA0001');

INSERT INTO Staff VALUES (0123460, 'Alan McCarthy', '02/02/1998', 'Duffy Hill, Enniscorthy, South Dublin', 'FaClAc01');

INSERT INTO Staff VALUES (0123461, 'Conor Doyle', '07/03/1981', '104 Creagh Ct., Rathcoole, Dublin', 'HeBe001');

INSERT INTO Staff VALUES (0123462, 'Mark Walsh', '16/06/1976', 'The Square, Oldcastle, South Dublin', 'GaToy01');

INSERT INTO Staff VALUES (0123463, 'Dave Smith', '19/10/1987', '10 Hanover Qy., Tel, Clondalkin, Dublin', 'HoFur123');

INSERT INTO Equipment VALUES ('HPT001', 'Hand pallet truck', 'Manual', 0123456);

INSERT INTO Equipment VALUES ('SEPT001', 'Semi-electric pallet truck', 'Semi-electric', 0123462);

INSERT INTO Equipment VALUES ('EPT001', 'Electric pallet truck', 'Electric', 0123459);

INSERT INTO Equipment VALUES ('MS001', 'Manual stacker', 'Manual', 0123461);

INSERT INTO Equipment VALUES ('SES001', 'Semi-electric stacker', 'Semi-electric', 0123457);

INSERT INTO Equipment VALUES ('ES001', 'Electric stacker', 'Electric', 0123463);

INSERT INTO Supplier VALUES (123456780, 'Sony Store', 'Sony Headphones', 23456789, '13:30', '+353876543210');

INSERT INTO Supplier VALUES (123456791, 'Amazon', 'Amazon Echo Dot', 23456790, '12:30', '+353876543211');

INSERT INTO Supplier VALUES (123456802, 'Tommy Hilfiger', 'Tommy Hilfiger Backpack', 23456791, '11:00', '+353876543212');

INSERT INTO Supplier VALUES (123456813, 'Perfeclan', 'Wooden Hand Drum', 23456792, '15:30', '+353876543213');

INSERT INTO Supplier VALUES (123456824, 'Elgato Store', 'Elgato Stream Deck', 23456793, '17:00', '+353876543214');

INSERT INTO Product VALUES (00001111, 'Sony Headphones', 1000, '€219,000', 123456780);

INSERT INTO Product VALUES (00001112, 'Amazon Echo Dot', 1000, '€29,990', 123456791);

INSERT INTO Product VALUES (00001113, 'Tommy Hilfiger Backpack', 1000, '€64,540', 123456802);

INSERT INTO Product VALUES (00001114, 'Wooden Hand Drum', 500, '€11,545', 123456813);

INSERT INTO Product VALUES (00001115, 'Elgato Stream Deck', 500, '€104,995', 123456824);

INSERT INTO Product\_Stock VALUES ('Elec0001', 00001111);

INSERT INTO Product\_Stock VALUES ('Elec0002', 00001112);

INSERT INTO Product\_Stock VALUES ('OutAcc0501', 00001113);

INSERT INTO Product\_Stock VALUES ('MuscDJ0007', 00001114);

INSERT INTO Product\_Stock VALUES ('HFHA0001', 00001115);

INSERT INTO Product\_Delivery VALUES (23456789, 12345678);

INSERT INTO Product\_Delivery VALUES (23456790, 12345679);

INSERT INTO Product\_Delivery VALUES (23456791, 12345680);

INSERT INTO Product\_Delivery VALUES (23456792, 12345681);

INSERT INTO Product\_Delivery VALUES (23456793, 12345682);